UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/614,953	07/08/2003	Young-Gook Ha	3364P116	7127

8791	7590	02/13/2006

EXAMINER
FABER, DAVID

BLAKELY SOKOLOFF TAYLOR & ZAFMAN
12400 WILSHIRE BOULEVARD
SEVENTH FLOOR
LOS ANGELES, CA 90025-1030

ART UNIT	PAPER NUMBER
2178	

DATE MAILED: 02/13/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

PTO-90C (Rev. 10/03)

	Application No.	Applicant(s)
Office Action Summary	10/614,953	HA ET AL.
	Examiner	Art Unit
	David Faber	2178

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on <u>08 July 2003</u>.

2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) <u>1-12</u> is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) <u>1-12</u> is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on <u>08 July 2003</u> is/are: a)☒ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☒ All b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____.

DETAILED ACTION

1.   This office action is in response to the application filed 8 July 2003.

This action is made Non-Final.

2.   Claims 1-12 are pending. Claims 1, 8, and 12 are independent.

Priority

3.   Acknowledgment is made of applicant's claim for foreign priority under 35

U.S.C. 119(a)-(d).

Information Disclosure Statement

4. The information disclosure statement filed 8 July 2003 fails to comply with the

provisions of 37 CFR 1.97, 1.98 and MPEP § 609 because on the IDS applicant

fails to list the Country Code for each of the Foreign Patent Document listed. In

addition, Document 1019950006911 is missing Page 1 of English Abstract of the

Foreign Patent Document. 37 CFR §1.98 of the MPEP states, "*Each foreign*

patent or published foreign patent application listed in an information disclosure

statement must be identified by the country or patent office which issued the

patent or published the application, an appropriate document number, and the

publication date indicated on the patent or published application." It has been

placed in the application file, but the information referred to therein has not been

considered as to the merits. Applicant is advised that the date of any re-

submission of any item of information contained in this information disclosure

statement or the submission of any missing element(s) will be the date of

submission for purposes of determining compliance with the requirements based

on the time of filing the statement, including all certification requirements for

statements under 37 CFR 1.97(e). See MPEP § 609.05(a).

5.   The information disclosure statement (IDS) submitted on 29 March 2005 is

in compliance with the provisions of 37 CFR 1.97. Accordingly, the information

disclosure statement is being considered by the examiner.

Drawings

6.   The drawings received on 8 July 2003 have been accepted by the

Examiner.

Claim Rejections - 35 USC § 112

7.   The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly
> claiming the subject matter which the applicant regards as his invention.

8.   Claims 1, 7, 8, 10, 11, and 12 are rejected under 35 U.S.C. 112, second

paragraph, as being indefinite for failing to particularly point out and distinctly

claim the subject matter which applicant regards as the invention. The Examiner

views the phrase, "grammar neutral document objects" as indefinite. For this

Office action, the Examiner views the phrase as ordinary objects used in

documents such as text.

Claim Rejections - 35 USC § 101

9.   35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or
> composition of matter, or any new and useful improvement thereof, may obtain a patent
> therefor, subject to the conditions and requirements of this title.

10.   Claims 1-8 rejected under 35 U.S.C. 101 because the claimed invention is

directed to non-statutory subject matter. The claims appear to be claiming

"software systems", which is a computer program per se. Since the computer

program is not embodied on a tangible computer readable medium, they appear

non-statutory.

11.   Claim 12 is rejected under 35 U.S.C. 101 because the claimed invention is

directed to non-statutory subject matter.

Section 2106 of the MPEP states:

(a) Functional Descriptive Material: "Data Structures" Representing Descriptive Material
Per Se or Computer Programs Representing Computer Listings Per Se
Data structures not claimed as embodied in computer-readable media are descriptive
material per se and are not statutory because they are not capable of causing functional
change in the computer. See, e.g., Warmerdam, 33 F.3d at 1361, 31 USPQ2d at 1760
(claim to a data structure per se held nonstatutory). Such claimed data structures do not
define any structural and functional interrelationships between the data structure and other
claimed aspects of the invention which permit the data structure's functionality to be
realized. In contrast, a claimed computer-readable medium encoded with a data
structure defines structural and functional interrelationships between the data structure
and the computer software and hardware components which permit the data structure's
functionality to be realized, and is thus statutory.
Similarly, computer programs claimed as computer listings per se, i.e., the descriptions or
expressions of the programs, are not physical "things." They are neither computer
components nor statutory processes, as they are not "acts" being performed. Such
claimed computer programs do not define any structural and functional interrelationships
between the computer program and other claimed elements of a computer which permit
the computer program's functionality to be realized. In contrast, a claimed computer-
readable medium encoded with a computer program is a computer element which defines
structural and functional interrelationships between the computer program and the rest of
the computer which permit the computer program's functionality to be realized, and is
thus statutory. Accordingly, it is important to distinguish claims that define descriptive
material per se from claims that define statutory inventions.
Computer programs are often recited as part of a claim. Office personnel should
determine whether the computer program is being claimed as part of an otherwise
statutory manufacture or machine. In such a case, the claim remains statutory irrespective
of the fact that a computer program is included in the claim. The same result occurs when
a computer program is used in a computerized process where the computer executes the
instructions set forth in the computer program. Only when the claimed invention taken as
a whole is directed to a mere program listing, i.e., to only its description or expression, is it

descriptive material per se and hence nonstatutory.

Since a computer program is merely a set of instructions capable of being executed by a computer, the computer program itself is not a process and **Office personnel should treat a claim for a computer program, without the computer-readable medium needed to realize the computer program's functionality, as nonstatutory functional descriptive material. When a computer program is claimed in a process where the computer is executing the computer program's instructions, Office personnel should treat the claim as a process claim.** See paragraph IV.B.2(b), below. When a computer program is recited in conjunction with a physical structure, such as a computer memory, Office personnel should treat the claim as a product claim.

Since the claim lacks an explicit statement that the computer program product is embodied in computer readable medium, it is viewed as non-statutory

Claim Rejections - 35 USC § 103

12.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

13.   Claims 1-12 are rejected under 35 U.S.C. 103(a) as being unpatentable over Cornelia et al (US Patent #6,065,026, patented 5/16/2000) further in view of Person et al (Person et al, "Special Edition Using Microsoft Word 97", published 12/16/1996, pp 1-15) in further view of Meyer, (Meyer, "aTool – Creating Validated XML Documents on the Fly Using MS Word," published 10/20/2002, pp 113-121).

As per independent Claim 1, Cornelia et al discloses a system comprising:

- a document component library for storing and managing document component summary information and document components that represent specific concepts; (Column 2, line 66 – Column 3, line 2: discloses a library used for storing components which are used to be assembled to create new documents. In addition, Column 9, lines 44-49, discloses Find Component menu option that is able to display a component dialog disclosing the component's name, description, author, text content, etc. Since the component contains all this information and displays it, the library stores components that contain summary information which represent specific concepts.)

- a document generation rule processor for accumulating document components needed for document assembly received from the document component library, and generating grammar neutral document objects (Column 20, lines 60 – 65: discloses creating documents by dragging and dropping language component icons where each icon represents a component. Documents are created by the dragging and dropping components into a list for the document thus creating non-grammar objects within a document. This process acts as document generation rule processor.)

However, Cornelia et al fails to specifically disclose a document generation rule formulator that a user employs to designate document generation rules through a graphic user interface. However, Person et al discloses Microsoft

Word contains templates, which contain parts of a document and features used for a specific type of document. (Page 1) Person et al discloses a form template, where assembly and context rules are used in which the assembly rules are used to create a table that creates a form while the context rules are used for font size and font type for the text. (FIG 6.2) In addition, Person et al discloses that all Word documents are based on a template wherein the default new document is based on a template containing default formatting and settings. (Page 5, lines 5-6) Thus, a user employs the designation of document rules when loading a template in Word. In addition, this process acts as a document generation rule formulator.

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to have combined Cornelia et al's method with Person et al's disclosure of templates in Microsoft Word since Person et al's disclosure of using a Microsoft Word template is a tool one could have used to further increase efficiency, productivity, and consistency when creating document.

In addition, Cornelia et al and Person et al fail to specifically disclose a document grammar connector for converting the grammar neutral document objects, which are suitable for processing in a program of a computer system, into grammar-connected documents in a string form used in an actual business. However, Meyer discloses using a tool extension to Microsoft Word (Abstract, lines 1-2, pg 1) that would create validated XML documents using Microsoft World. (Title, pg 1) Meyer's tool would convert the MS Word document into XML generating a grammar-connected document (Page 116, Right Column, lines 30-

37) by making sure the document is valid and complaint with its DTD (Page 114,

Right Column, Lines 30-31). This process acts as a document grammar

connector.

It would have been obvious to one of ordinary skill in the art at the time of

applicant's invention to have combined Cornelia et al's and Person et al's

methods with Meyer's aTool method since it would have provided the aTool as a

hybrid solution that offers the benefits of MS Word with the costs of a little less

XML support.

As per dependent Claim 2, Cornelia et al discloses:

- a component selector for displaying usable component items that are

    provided by a corresponding library based on document component

    summary information searched in the document component library, the

    document component summary information including at least a

    component ID, a component name, and a component type, and

    optionally including various different types of information that represent

    other components; (Column 9, lines 42-52: Discloses able to using a

    Find Component menu option to find components based on the

    component's information such as the component's name in which the

    system returns with a list of components in the library based on the

    component's information searched. The user is able to insert

    components into the document from the component list.)

- a document component assembler for forming an area where

    component structures are modeled based on user input through a

graphic user interface, the user dragging the needed document components appearing in the component selector and dropping the documents at a suitable location in the document component assembler to thereby generate document structures (Column 20, line 60 – Column 21, line 4: Discloses documents being created by dragging and dropping components into a list for the document using a tree viewer. Once the list been created, the word document with complete content is generated by a user action. This creation creates a structured document containing components placed in a structural manner.

However, Cornelia et al fails to specifically disclose which such structures are formulated as assembly rules, and the assembly rules include IDs of all document components and structural information between each component; and disclose a context condition compiler for forming an area where context conditions realized through pairs of conditions and actions are compiled to enable insertion into document structures, the context condition compiler enabling the formulation of context rules, which allow the processing of actions, in the document generation rule processor in the case where conditions are satisfied for a specific business context during document assembly. However, However, Person et al discloses Word contains templates, which contain parts of a document and features used for a specific type of document. (Page 1) Person et al discloses in FIG. 6.2 using Microsoft Word where a template is showing structural information where a table has been implemented to produce a form

layout, and additional structural information created to divide the table into cells where text has been place. Thus, FIG 6.2 discloses assembly rules used to create a form layout contain structural information to properly place components for table and text where it was assembled. In addition, Person et al's discloses where context conditions are used by the formulation rules where the template created allowing the user to easily enter information by requiring only the user just to point, click, and type to fill out a form. (Page 4, Paragraph 4; FIG 6.4)

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to have combined Cornelia et al's method with Person et al's disclosure since Person et al's disclosure of using a Microsoft Word template is a tool one could have used to further increase efficiency, productivity, and consistency while reducing company's costs when creating a document.

In addition, Cornelia et al discloses the author has the ability to get a listing of all the components in the document that displays component identifiers such as the component's name. (Column 11, lines 51-58) Since Person et al discloses Microsoft Word is able to assembled structural information or rules, Word would been able to include the identifiers using the Show Document Components menu option from Cornelia et al's application since Cornelia et al's application is built using Microsoft Word. (Column 6, lines 22-47)

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to have combined Cornelia et al's and Person et al's method with the use of identifiers since it would have allowed using computers to assemble documents to reduce the amount of time that attorneys and other

individuals who prepare long legal or transactional documents spend on the

mechanics of document preparation.

As per dependent Claim 3, Cornelia et al fails to specifically disclose the

assembly rules and the context rules are output as a single document generation

rule. However, Person et al discloses Word contains templates, which contain

parts of a document and features used for a specific type of document. (Page 1)

Person et al discloses a specific type of a form template, where assembly and

context rules are used in which the assembly rules are used create a table that

creates a form while the context rules are used for font size and font type for the

text. (FIG 6.2) thus a template is a single document generation rule involving the

combination of assembly and context rules.

It would have been obvious to one of ordinary skill in the art at the time of

applicant's invention to have combined Cornelia et al's method with Person et

al's disclosure of templates since Person et al's disclosure of using a Microsoft

Word template is a tool one could have used to further increase efficiency,

productivity, and consistency while reducing company's costs when creating a

document.

As per dependent Claim 4, Cornelia et al discloses a system:

- the document component summary information for recording various

  document components that constitute business documents and

  detailed information on all components included in a present library;

  and ; (FIG 26; Column 12, line 65 – Column 13, line 1: Discloses the

  ability to view one component from the library which discloses detailed

component information that is used for business purposes (Column 6,

lines 43-47)

- a component library interface for connection to external modules, (FIG.

  3; Column 5, lines 27-38: Discloses the word processor, as a separate

  module, having an API that interacts with the library object which

  interacts with the library.)

- the document generation rule formulator searches the document

  component summary information through the component library

  interface, and the document generation rule processor uses document

  component Ids to accumulate document components required for

  document assembly. (Column 6, lines 6-12 discloses the interaction

  using the library. This disclosure enables using a Find Component

  menu option to find components based on the component's information

  such as the component's name in which the system returns with a list

  of components in the library based on the component's information

  searched then allowing the author to insert the component into

  document thus outputting onto the document. (Column 9, lines 42-52))

However, Cornelia et al fails to specifically discloses the component Ids

are numbers specific to each component. However, it was well-known to one of

ordinary skill in the art at the time of applicant's invention that a number can be a

name and that identifiers were programmed as/into numbers within a data

structure .

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to have combined Cornelia et al's method with the disclosure of numbers used as name as identifiers since it would have allowed a user an easier method for storing data within a recording medium.

As per dependent Claim 5, Cornelia et al discloses a system:

- wherein the document components stored in the document component library include simple components of a single type and complex components realized through a structure of a plurality of simple components. (Column 2, line 66 – Column 3, line 2 discloses a library containing components which are unit of text that is shared among documents. A component may contain variable text in order to facilitate flexibility and foster re-use. (Column 2, lines 48-50) Column 10, lines 8-34 discloses adding variable text into the component.)

As per dependent Claim 6, Cornelia et al discloses a system:

- a component assembler for using document component IDs to accumulate from the document component library the document components required in the assembly rule, resulting assembled components are outputted. (Column 9, lines 42-52: Discloses using a Find Component menu option to find components based on the component's information such as the component's name in which the system returns with a list of components in the library based on the

component's information searched then allowing the author to insert

the component into document thus outputting onto the document.)

Cornelia et al fails to specifically disclose assembling the document

components using structural information between components and reading

assembly and a context processor for reading context rules in the document

generation rules, and if a specific business context satisfies the conditions of the

context rules, applying designated actions to the assembled components to

thereby ultimately generate the grammar neutral document objects. However,

Person et al discloses in FIG. 6.2 using Microsoft Word where a template is

showing structural information where a table has been implemented to produce a

form layout, and additional structural information created to divide the table into

cells where text has been place. Thus, FIG 6.2 discloses assembly rules used to

create a form layout contain structural information to properly place components

for table and text where it was assembled. In addition, Person et al's discloses

where context conditions are used by the formulation rules where the template

created allowing the user to easily enter information by requiring only the user

just to point, click, and type to fill out a form. (Page 4, Paragraph 4; FIG 6.4) This

process acts as a context processor that discloses an embodiment using a

template showing context conditions allowing the user only have to point, click,

and type information in a already constructed form by the template's assembly

and context rules.

It would have been obvious to one of ordinary skill in the art at the time of

applicant's invention to have combined Cornelia et al's method with Person et

al's disclosure since Person et al's disclosure of using a Microsoft Word template is a tool one could have used to further increase efficiency, productivity, and consistency while reducing company's costs when creating a document.

As per dependent Claim 7, Cornelia et al discloses a document output unit saving final documents to the library after creating a document with components (Column 15, lines 49-57) for business uses (Column 6, lines 43-47) However, Cornelia et al and Person et al fail to specifically disclose a grammar converter supporting grammar for specific business systems and converting the grammar neutral document objects into grammar-connected document objects;

However, Meyer discloses using a tool extension to Microsoft Word that would create validate XML documents using Microsoft World. Meyer's tool would convert the MS Word document into XML generating a grammar-connected document (Page 116, Right Column, lines 30-37) by making sure the document is valid and in compliance with its DTD (Page 114, Right Column, Lines 30-31). An XML document is inherently considered as a recognizable string format by the user.

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to have combined Cornelia et al's and Person et al's methods document generation with Meyer's aTool method to create XML document from Word that would have provided the aTool as a hybrid solution that offers the benefits of MS Word with the costs of a little less XML support.

As per independent Claim 8, Roberts et al discloses a method comprising:

- (a) storing document component summary information and document

    components that represent specific concepts; (Column 2, line 66 –

    Column 3, lines 1-2: discloses a library storing components used to be

    assembled to create new documents. In addition, Column 9, lines 44-

    49, discloses a option within Cornelia et al's application of a Find

    Component function that able to display a component dialog disclosing

    the component's name, description, author, text content etc. Since the

    component contains all this information and displays it, the library

    stores components that contain summary information which represent

    specific concepts.)

- (c) accumulating document components needed for document

    assembly and from a document component library, and generating

    grammar neutral document objects based on the document generation

    rules', and (Column 20, lines 60 – 65: discloses creating documents

    by dragging and dropping language component icons where each icon

    represents a component. Documents are created by the dragging and

    dropping components into a list for the document thus creating non-

    grammar objects within a document.)

However, Cornelia et al fails to specifically disclose designating document

generation rules through a graphic user interface. However, Person et al

discloses Microsoft Word contains templates, which contain parts of a document

and features used for a specific type of document. (Page 1) Person et al

discloses a form template, where assembly and context rules are used in which the assembly rules are used to create a table that creates a form while the context rules are used for font size and font type for the text. (FIG 6.2)   In addition, Person et al discloses that all Word documents are based on a template wherein the default new document is based on a template containing default formatting and settings. (Page 5, lines 5-6) Thus, the designation of document rules occurs when loading a template in Word.

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to have combined Cornelia et al's method with Person et al's disclosure of templates since Person et al's disclosure of using a Microsoft Word template is a tool one could have used to further increase efficiency, productivity, and consistency while reducing company's costs when creating a document.

In addition, Cornelia et al and Person et al fail to specifically disclose converting the grammar neutral document objects, which are suitable for processing in a program of a computer system, into grammar-connected documents in a string form used in an actual business. However, Meyer discloses using a tool extension to Microsoft Word that would create validate XML documents using Microsoft World. Meyer's tool would convert the MS Word document into XML generating a grammar-connected document (Page 116, Right Column, lines 30-37) by making sure the document is valid and compliance with its DTD (Page 114, Right Column, Lines 30-31). An XML document is inherently considered as a recognizable string format by the user.

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to have combined Cornelia et al's and Person et al's methods of document generation with Meyer's aTool method of creating XML documents from Microsoft Word would have provided the aTool as a hybrid solution that offers the benefits of MS Word with the costs of a little less XML support.

As per dependent Claim 9, Cornelia et al discloses a method:

- displaying a list of usable components provided library based on the by a corresponding document component summary information searched in the document component library; (Column 9, lines 42-52: Discloses using a Find Component menu option to find components based on the component's information in which the system returns with a list of components in the library based on the component's information searched.)

- dragging required documents appearing in a component selector and dropping the documents at a suitable location in a document component assembler, which forms an area where component structures are modeled based on user input through a graphic user interface, to thereby generate document structures; and (Column 20, line 60 – Column 21, line 4: Discloses documents being created by dragging and dropping components into a list for the document using a tree viewer. Once the list been created, the word document with complete content is generated by a user action. This creation creates a

structured document containing components placed in a structural

manner.)

However, Cornelia et al fails to specifically disclose compiling context

conditions realized through pairs of conditions and actions, and allowing insertion

of the context conditions into document structures. However, Person et al

discloses Word contains templates, which contain parts of a document and

features used for a specific type of document. (Page 1) However, Person et al

discloses in FIG. 6.2 using Microsoft Word where a template is showing

structural information where a table has been implemented to produce a form

layout, and additional structural information created to divide the table into cells

where text has been place. Thus, FIG 6.2 discloses assembly rules used to

create a form layout contain structural information to properly place components

for table and text where it was assembled. In addition, Person et al's discloses

where context conditions are used by the formulation rules where the template

created allowing the user to easily enter information by requiring only the user

just to point, click, and type to fill out a form. (Page 4, Paragraph 4; FIG 6.4)

It would have been obvious to one of ordinary skill in the art at the time of

applicant's invention to have combined Cornelia et al's method with Person et

al's disclosure of templates since Person et al's disclosure of using a Microsoft

Word template is a tool one could have used to further increase efficiency,

productivity, and consistency while reducing company's costs when creating a

document.

As per dependent Claim 10, Cornelia et al discloses:

- accumulating from the document component library the document
  components required in the assembly rules using document
  component Ids, and outputting the resulting assembled components.
  (Column 9, lines 42-52: Discloses using a Find Component menu
  option to find components based on the component's information such
  as the component's name in which the system returns with a list of
  components in the library based on the component's information
  searched then allowing the author to insert the component into
  document thus outputting onto the document.)

Cornelia et al fails to specifically disclose assembling the document
components using structural information between components and reading
assembly and context rules in the document generation rules, and if a specific
business context satisfies the conditions of the context rules, applying designated
actions to the assembled components to thereby ultimately generate the
grammar neutral document objects. However, Person et al discloses Word
contains templates, which contain parts of a document and features used for a
specific type of document. (Page 1) Person et al discloses in FIG. 6.2 using
Microsoft Word where a template is showing structural information where a table
has been implemented to produce a form layout, and additional structural
information created to divide the table into cells where text has been place. Thus,
FIG 6.2 discloses assembly rules used to create a form layout contain structural
information to properly place components for table and text where it was

assembled. In addition, Person et al's discloses where context conditions are used by the formulation rules where the template created allowing the user to easily enter information by requiring only the user just to point, click, and type to fill out a form. (Page 4, Paragraph 4; FIG 6.4) This process acts as a context processor that discloses an embodiment using a template showing context conditions allowing the user only have to point, click, and type information in a already constructed form by the template's assembly and context rules.

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to have combined Cornelia et al's method with Person et al's disclosure of templates since Person et al's disclosure of using a Microsoft Word template is a tool one could have used to further increase efficiency, productivity, and consistency while reducing company's costs when creating a document.

As per dependent Claim 11, Cornelia disclose saving final documents to the library after creating a document with components (Column 15, lines 49-57) for business uses (Column 6, lines 43-47) However, Cornelia et al and Person et al fail to specifically disclose supporting grammar for specific business systems and converting the grammar neutral document objects into grammar-connected document objects.

However, Meyer discloses using a tool extension to Microsoft Word that would create validate XML documents using Microsoft World. Meyer's tool would convert the MS Word document into XML generating a grammar-connected document (Page 116, Right Column, lines 30-37) by making sure the document

is valid and complaint with its DTD (Page 114, Right Column, Lines 30-31). An XML document is inherently considered as a recognizable string format by the user.

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to have combined Cornelia et al's and Person et al's methods of document generation with Meyer's aTool method of creating XML documents from Microsoft Word would have provided the aTool as a hybrid solution that offers the benefits of MS Word with the costs of a little less XML support.

As per independent Claim 12, Claim 12 recites similar limitations as in Claim 8 and is similarly rejected under Cornelia et al, Person et al, and Meyer.

Conclusion

14.   The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

- Lindhorst et al (US Patent #6,714,219): Discloses drag and drop of objects into a position in a document.

- Hanson et al (US Patent #5,956,736): Discloses a object-oriented editor for creating web documents.

- Hill (Hill, "Microsoft Word Templates," pg 1-9): Discloses information regarding Microsoft Word Templates.

- Dodge et al (US Patent #5,655,130): Discloses document production using common document database.

- Roberts et al (US PGPub 2002/0143818): Discloses a system for creating structured documents using an graphical interface and the use of DTD.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to David Faber whose telephone number is 571-272-2751. The examiner can normally be reached on M-F from 8am to 430pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Stephen Hong, can be reached on 571-272-4124. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

David Faber
Patent Examiner
AU 2178

STEPHEN HONG
SUPERVISORY PATENT EXAMINER